

MODELOS DE PROCESO PARA LA INTEGRACIÓN DEL NEGOCIO UTILIZANDO SERVICE ORIENTED ARCHITECTURE (SOA).

López, G.¹; Jeder, I.¹; Echeverría, A.¹; Fierro, P. (PhD.)²

1. Laboratorio de Informática de Gestión
Facultad Ingeniería - Universidad de Buenos Aires
Tel: 54-11-4343-0891, Extensión 141
e-mail: glopez@fi.uba.ar
Web: <http://www.fi.uba.ar/laboratorios/lig>

2. Departamento de Estudios e Investigación de la
Empresa - Facultad de Economía - Universidad de
Salerno, Italia
Tel: +390815519623
e-mail: fierrop@unisa.it
<http://www.unisa.it>

1. ANTECEDENTES

La Arquitectura Orientada a Servicios es el siguiente paradigma arquitectónico luego del advenimiento del modelo cliente/servidor y las aplicaciones basadas en Internet. La diferencia con esos paradigmas es que en lugar de focalizarse en la tecnología, hace foco en cómo organizar los negocios. Visto del lado del negocio, una arquitectura orientada a servicios es una técnica de desarrollo de aplicaciones de alto nivel que le permite a la IT focalizar en los procesos de negocio, antes que en la infraestructura, para alcanzar una ventaja competitiva. Las industrias que implementen satisfactoriamente SOA, seguramente poseerán una ventaja competitiva importante sobre las industrias que no, porque las que tienen sus servicios alineados con los negocios estratégicos de IT, pueden reaccionar más rápido a los cambios en los requerimientos de negocio que las que no los tienen. SOA es una arquitectura en la cual una aplicación se constituye de servicios que se exponen y servicios que se consumen; difiere del tradicional enfoque cliente/servidor haciendo énfasis en el bajo acoplamiento entre los componentes de software. Dado que estos servicios pueden ser consumidos por diferentes sistemas y plataformas, las características de los Web Services son ideales para implementar esta solución, pero debe seguirse algún tipo de Modelo de Ciclo de Vida para el desarrollo de aplicaciones con esta arquitectura.

1.1 DISEÑO DE SERVICIOS. El uso de los principios del diseño orientado a servicios en la lógica de procesamiento da como resultado una lógica de procesamiento orientada a servicios.

Los principales principios de una arquitectura orientada a servicios son los siguientes:

- **Desacoplamiento:** Los servicios mantienen una relación que minimiza las dependencias y solo requiere que estos conozcan la existencia de los otros.
- **Contrato de servicio:** Los servicios se adhieren a un acuerdo de comunicación el cual está definido por las descripciones de servicios.
- **Autonomía:** Los servicios tienen el control sobre la lógica que ellos encapsulan.
- **Abstracción:** Los servicios ocultan su lógica de procesamiento. Solo especifican lo descrito en el contrato de servicio.
- **Reutilización:** La lógica se encuentra dividida en servicios con la intención de hacer reutilizarla.
- **Composición:** Una colección de servicios puede ser coordinada y unida de forma tal de componer servicios.
- **Ausencia de estado:** Los servicios minimizan la información retenida sobre una tarea específica.
- **Descubrimiento:** Los servicios se encuentran diseñados de forma tal de poder ser descriptos con el objeto de poder ser encontrados y evaluados a través de mecanismos de descubrimiento.

2 PROBLEMÁTICA

La utilización de servicios permite introducir modificaciones en las aplicaciones sin la necesidad de invertir grandes cantidades de tiempo y dinero. Se consigue de esta forma un

enorme valor al poder optimizar y mejorar continuamente los procesos a medida que las estrategias del negocio cambian.

No existe un Modelo de Ciclo de Vida ad-hoc para la integración de los procesos de negocio en aplicaciones con arquitectura orientada a servicios (SOA); este trabajo intenta ser un aporte en tal sentido ya que el desarrollo de aplicaciones con Arquitectura Orientada a Servicios requiere de la aplicación de modelos propios. Los Ciclos de Vida se encuentran formados por una serie de fases, etapas o pasos requeridos para obtener una solución SOA a partir de un conjunto de necesidades determinado.

El propósito de todo ciclo de vida es planear, ejecutar y controlar el proyecto de desarrollo de un sistema. El ciclo de vida define las fases y las tareas esenciales para el desarrollo de sistemas, sin importar el tipo o la envergadura del sistema que se intenta construir.

No existe un Modelo de Ciclo de Vida que sirva para cualquier proyecto; cada proyecto debe seleccionar un ciclo de vida que sea el más adecuado para su caso.

3 SOLUCIÓN PROPUESTA

Modelos de Ciclos de Vida para desarrollo de proyectos con Arquitectura Orientada a Servicios

Si bien los Modelos de Ciclo de Vida para proyectos SOA se basan en los pilares de los ciclos de vida para soluciones distribuidas, los mismos requieren de ciertas adaptaciones para obtener un producto de calidad.

3.1 MCVS SOA con enfoque Top – Down.

La arquitectura resultante con este enfoque será óptima debido a que se comienza analizando el flujo de negocio de manera integral, para luego bajar el nivel de detalle hasta los servicios a implementar. La principal desventaja de este enfoque es el presupuesto y tiempos.

Las fases que integran este MCVS aplicado a proyectos SOA, son las siguientes:

3.1.1 Análisis. En esta primera fase se determina el alcance del proyecto SOA. Antes de modelar el esquema de servicios se comienza por analizar en detalle el flujo y reglas de negocio de la organización. Luego surgen los principales servicios candidatos, y se definen las capas a utilizar.

3.1.2 Diseño. Una vez definido el análisis, se puede comenzar a diseñar de qué forma implementarlo. Esta fase es por lo general dirigida en base a estándares e incorpora principios y convenciones establecidas para sistemas orientados a servicios.

3.1.3 Desarrollo. Determinadas las tecnologías sobre las cuales se construirán los componentes de la arquitectura orientada a servicios, sólo basta construirlos. Existe una gran diversidad de tecnologías, herramientas y plataformas para simplificar el desarrollo del proyecto.

3.1.4 Pruebas. Debido a que los servicios serán potencialmente módulos reutilizables en una gran variedad de escenarios, su calidad debe ser rigurosamente controlada.

3.1.5 Implantación. En base a las tecnologías específicas seleccionadas para el desarrollo se podrán definir las actividades que formen parte de la fase de implantación.

3.1.6 Administración. La naturaleza de los factores a administrar para este tipo de sistemas va a ser muy similar a la utilizada para sistemas distribuidos basados en componentes.

La administración incluye el monitoreo de servicios, control de versiones, seguimiento de mensajes, detección de cuellos de botella.

3.2 MCVS SOA con enfoque Bottom-Up

El enfoque bottom-up establece una perspectiva diferente durante el análisis. El mismo propone comenzar a construir los servicios a partir de requerimientos puntuales, como por ejemplo, establecer canales de integración punto a punto entre sistemas, o reemplazar soluciones de comunicación remota de aplicaciones por un protocolo multiplataforma como SOAP (Simple

Object Access Protocol). Muchas veces estos requerimientos pueden resolverse simplemente implementando servicios sobre módulos de un sistema ya existente.

Las organizaciones podrían ver ventajoso a este modelo ya que les permite integrar sus sistemas utilizando nuevas tecnologías a bajo costo. A pesar de que las implementaciones de este tipo podrían resultar exitosas, es decir, lograr su objetivo de integración puntual, no se encontrarían enmarcadas en una arquitectura diseñada para aprovechar la Orientación a Servicios en su máxima expresión. Las soluciones desarrolladas bajo este Modelo no están concebidas para soportar un gran número de servicios de forma consistente, robusta y ágil.

3.3 MCVS SOA con enfoque Ágil

Con la finalidad de encontrar un enfoque que permita incorporar los principios de arquitectura orientada a servicios en los ambientes de negocio, sin necesidad de esperar que se haya finalizado el proceso en toda la organización, ha surgido el MCVS con enfoque ágil.

La modalidad de trabajo de este modelo difiere de las anteriores ampliamente ya que se ocupa de ejecutar el análisis del negocio en paralelo al diseño de servicios y desarrollo.

Esta forma de trabajo tiene una componente de esfuerzo adicional, con el lógico costo asociado. Esto se debe a la necesidad de tener que ajustar los servicios construidos para alinearlos con los modelos de negocio que pueden ir cambiando a medida que se avanza con el análisis.

Las fases que integran este MCVS aplicado a proyectos SOA, son las siguientes:

3.3.1 Análisis

La fase de análisis debe focalizar en el modelo de negocio. En el momento en que se tiene suficiente conocimiento de las áreas del negocio, se va a comenzar con el trabajo en paralelo de modelado de servicios de negocio. Este punto de inflexión, denominado el punto de maduración del análisis de negocio, debe ser determinado apelando al sentido común y experiencia. Si es muy temprano para comenzar con el modelado de servicios, seguramente se requerirá un trabajo de reingeniería para adaptar los servicios al modelo final de negocio, y si por el contrario, se espera demasiado para comenzar por los servicios, se estará perdiendo la agilidad que podría destacar a este modelo frente a los dos precedentes.

3.3.2 Diseño, Desarrollo, Pruebas e Implantación. Estas fases van a ser ejecutadas en paralelo a la etapa de análisis a medida que nuevos servicios se incorporan a la arquitectura. Se trabaja con el diseño de cada componente de servicio que surge del análisis. Luego se desarrollan, prueban e implantan los servicios diseñados.

3.3.3 Revisión. Se deben efectuar revisiones periódicas de la arquitectura actual contra los modelos de negocio obtenidos. A partir de estas revisiones, cuya intención es encontrar inconsistencias entre la implementación y la realidad, surgirán planes de ejecución de adaptaciones de los servicios construidos para alinearlos con las necesidades actuales.

Cada servicio que deba ser modificado, tendrá que pasar por las etapas de diseño (o mejor dicho re-diseño), desarrollo, pruebas e implantación nuevamente.

3.4 MCVS con enfoque RUP + XP

Existe una tendencia por la adopción de una metodología mixta, la cual toma una u otra forma dependiendo del momento del ciclo de vida en el cual se encuentra el proyecto SOA.

Un proyecto SOA consta de dos grandes fases de alto nivel [Mittal, 2006]. La primera es la construcción de la plataforma SOA. Esta fase requiere la utilización de una metodología como RUP (Racional Unified Process). La segunda fase es la de mantenimiento, en la cual nuevos proyectos son construidos sobre la arquitectura inicial. En esta fase se pueden aprovechar las ventajas de una metodología liviana como XP (eXtreme Programming).

3.5 MCVS SOA con enfoque de Gobierno constante

Este modelo consiste en un conjunto de fases que son ejecutadas de forma iterativa, proporcionando una mejora continua del proceso.

3.5.1 Modelado. Durante esta fase se obtienen y analizan los requerimientos de negocio con el fin de llegar a un modelo de procesos de negocio que será la base para el diseño de servicios y niveles de servicio. Estos procesos también serán útiles a la hora de medir la eficiencia del negocio. Es de gran importancia que en esta etapa se fije un modelo común que sea comprendido tanto por gente de IT como de negocio.

3.5.2 Ensamble. Una vez que se han definido los procesos de negocio, se deben obtener los servicios necesarios para que los mismos puedan ser incorporados a la solución. Para esto puede ser necesario la construcción de nuevos servicios, utilizar servicios ya existentes, o encapsular funcionalidades de sistemas existentes. Por último, se utilizará un mecanismo de orquestación de servicios que permita que los procesos de negocio cobren vida.

3.5.3 Implantación. En esta etapa se debe configurar el ambiente de ejecución de los servicios para lograr cumplir con los niveles de calidad fijados y así poder ejecutar exitosamente los procesos de negocio. Es fundamental que el ambiente de servicios sea robusto, escalable y seguro. Este ambiente debe estar preparado tanto para correr procesos de misión-crítica como para aceptar cambios de forma flexible.

3.5.4 Administración. La fase de Administración incluye establecer y mantener la disponibilidad de los servicios y sus tiempos de respuesta. Se deben monitorear los KPI (Key Performance Indicators) en tiempo real para prevenir, aislar, diagnosticar y solucionar problemas. Es también una tarea a llevar a cabo en esta etapa la de administrar y mantener un control de versiones sobre los servicios que corren los procesos de negocio.

3.5.5 Gobierno y Procesos. Este proceso debe ser ejecutado durante todo el ciclo de vida. Se deben establecer políticas y procesos que aseguren el éxito del proyecto SOA. Por ejemplo, se puede crear un centro de excelencia para implementar políticas de gobierno y controlar que los estándares se cumplan.

4 CONCLUSIONES

La arquitectura de un sistema es la organización o estructura de sus partes o componentes más importantes, que permite que los involucrados tengan una visión común del sistema completo. La arquitectura va es un elemento fundamental para la toma de decisiones durante el proyecto. Se debe tener en cuenta el factor de calidad, rendimiento, reutilización y capacidad de evolución. Las ventajas y beneficios del uso de una Arquitectura Orientada a Servicios con web services incluyen un mejor ROI para los proyectos, resultados más rápidos y capacidad para responder ágilmente a los cambios del negocio. Estas ventajas competitivas estarán presentes toda vez que el proyecto haya sido concebido siguiendo un Modelo de desarrollo adecuado. No todos los modelos sirven para todos los proyectos SOA.

Es necesario tener en cuenta que SOA es una evolución y no una revolución; que no cambiará las tecnologías existentes sino que las integrará. Como cualquier iniciativa exitosa se debe desarrollar un plan estratégico para alcanzar los requerimientos de negocio asociados.

Como fue planteado en el acápite de Problemática, si bien no existe un Modelo de Ciclo de Vida ad-hoc para la integración de los procesos de negocio en aplicaciones con arquitectura orientada a servicios (SOA), pueden definirse las etapas de Análisis y Diseño con tareas propias y específicas de esta arquitectura, a saber:

- Análisis orientado a servicios

El proceso de determinar como la automatización de los requerimientos del negocio pueden representarse por medio de servicios es la tarea del análisis orientado a servicios.

Durante esta etapa se definen:

- Los servicios que son necesarios construir.
- Que lógica debe ser encapsulada por casa servicio.

Objetivos:

- Definir un grupo preliminar de servicios.
- Definir un límite preliminar de los servicios, de forma tal de no reescribir un servicio existente o ya planeado.
- Identificar la lógica encapsulada haciendo hincapié en la reutilización.
- Definir los modelos conocidos de composición de servicios.

El análisis orientado a servicios no necesariamente implica reinventar los procedimientos existentes dentro de una organización en materia de análisis de procesos. Más bien, propone una serie de pasos a seguir para complementar estos procedimientos en miras de realizar un diseño orientado a servicios.

○ Diseño orientado a servicios

En este proceso se generan diseños concretos de los servicios a partir de los servicios candidatos identificados en la etapa anterior. Estos servicios son luego ensamblados en una composición abstracta que implementa un proceso de negocio.

Durante esta etapa se determina:

- Definición de las interfaces de servicio
- Características de SOA a implementar
- Estándares y extensiones requeridas

Objetivos:

- Determinar el grupo principal de extensiones arquitecturales.
- Determinar los límites de la arquitectura.
- Identificar los estándares de diseño necesarios
- Definir las interfaces de los servicios
- Identificar las posibles composiciones de servicio

6 BIBLIOGRAFÍA

- Erl, T. "Service Oriented Architecture Concepts Technology And Design". Prentice Hall. ISBN 0-13-185858-0. Agosto 2004.
- Fitzgerald, B. The system development dilemma: whether to adopt formalized systems development methodologies or not? In Baets, W.(ed.) Proceedings of the Second European Conference on Information Systems, Nijenrode University Press, Holland, 691-706. 1994.
- Fitzgerald, B. The Use of Systems Development Methodologies in Practice: A field Study, The Information Systems Journal, Vol. 7, No 3, pp. 201-212. 1997.
- IBM. "IBM SOA Foundation: providing what you need to get started with SOA". White paper. Septiembre 2005.
- IEEE. 1991. Standard for Developing Software Life Cycle Processes. IEEE Std. 1074-1991, Nueva York, IEEE Computer Society.
- Newcomer, E., Lomow G.. "Understanding SOA with Web Services (Independent Technology Guides)". Addison-Wesley Professional. ISBN 0321180860. Diciembre 2004.
- Pulier E., Taylor H.. "Understanding Enterprise SOA". Noviembre 2005.
- Woods, D., Mattern, T.. 2006. Enterprise SOA: Designing IT for Business Innovation. O'Reilly Press.